МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики

Мегафакультет трансляционных информационных технологий

Факультет информационных технологий и программирования

**Лабораторная работа №4**

**По дисциплине «Web-программирование»**

**Создание контроллеров страниц и спецификации**

**Выполнили студент группы М33081**

**Аль Даббагх Харит Хуссейн**

**Проверил**

**Приискалов Роман Андреевич**

САНКТ-ПЕТЕРБУРГ

2022

Содержание

[Module Creation 2](#_Toc99875781)

[Generating modules 2](#_Toc99875782)

[Swagger OpenAPI 4](#_Toc99875783)

# Module Creation

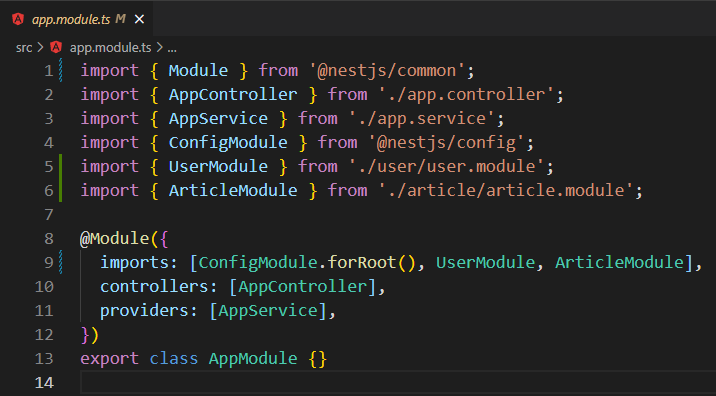
## Generating modules

In order to generate a module for our models, we can use the following commands:

nest g module user

nest g module article

This will create a module for our user model and another for the article model. Then it will automatically import these modules in our base app module.



Modules serve as a way to organize our code especially when we’ll have a lot of entities and team members working on the same project.

For our API to work we also need to create controllers and services, which will then be imported in their corresponding modules.

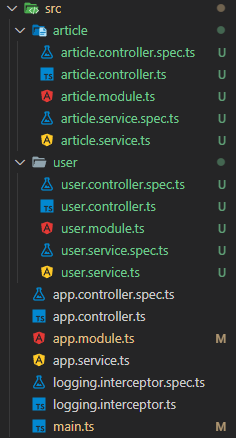
To do that from the CLI:

nest g controller user

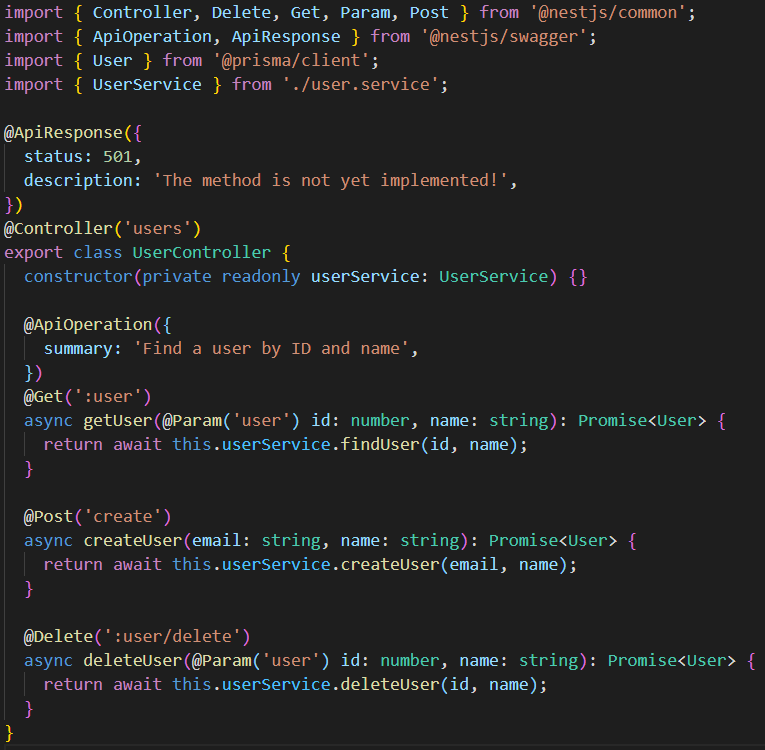
nest g controller article

nest g service user

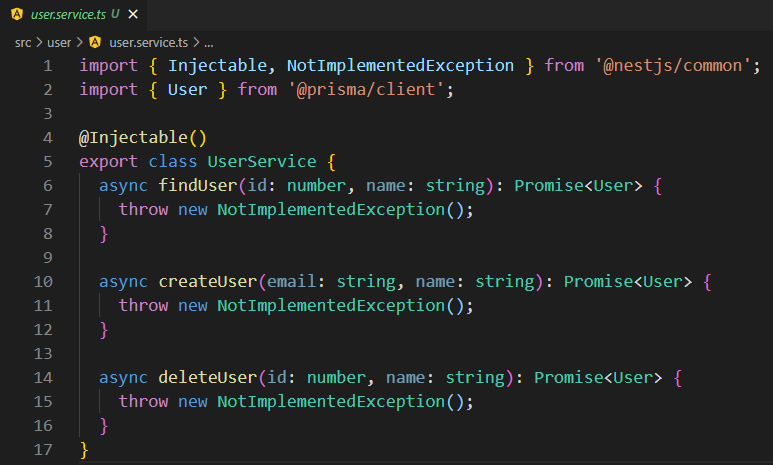
nest g service article

The result file heirarchy should look something like in the right screenshot:

Next we need to create our endpoints (controller) for each of these models. The implementation for the user model should look something like this:



The same way we need to create the service for that model (In which the business logic will be implemented in the future). An example for the user service should look like this:

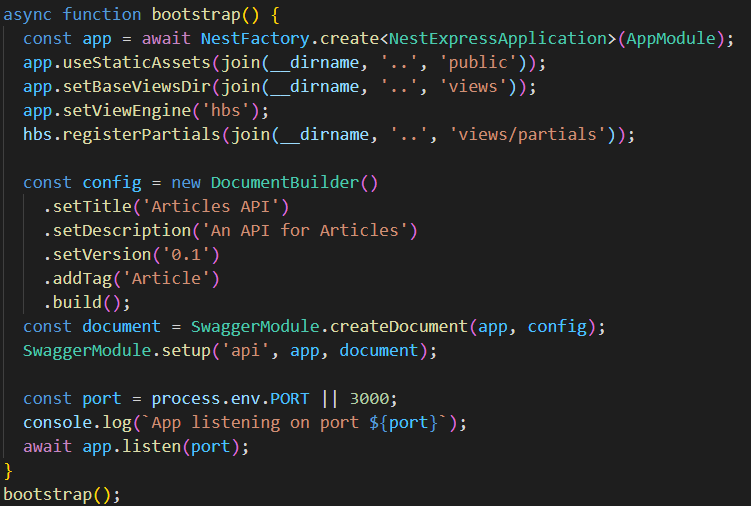


## Swagger OpenAPI

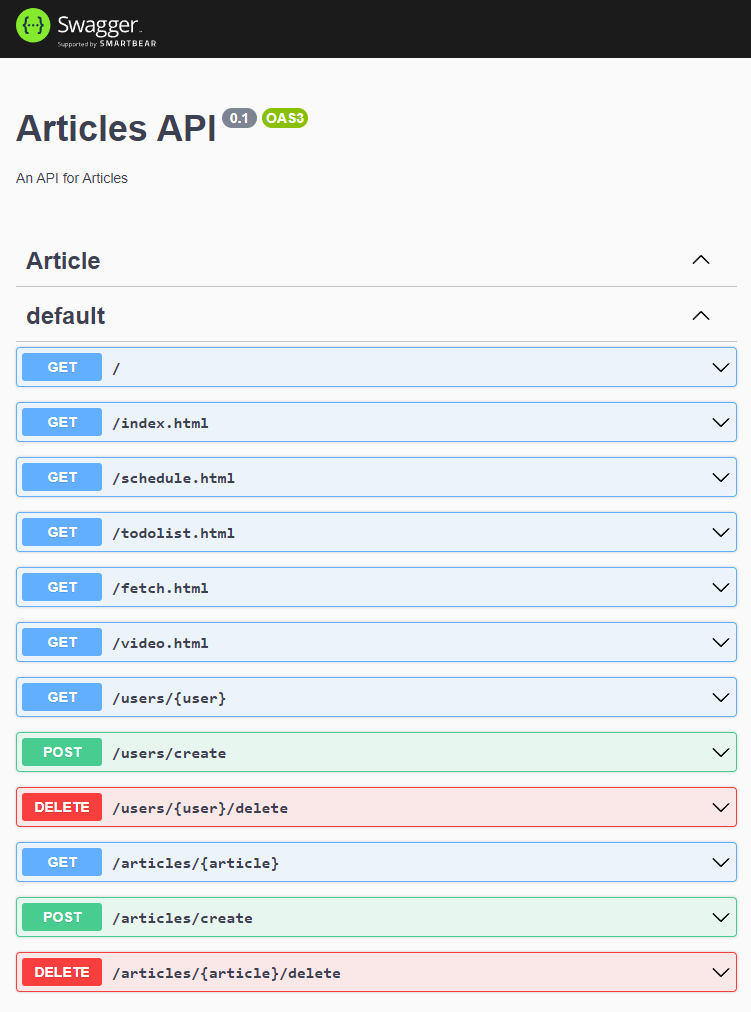
Swagger is a way for us to document and easily test our API. It needs to be installed using the command:

npm install --save @nestjs/swagger swagger-ui-express

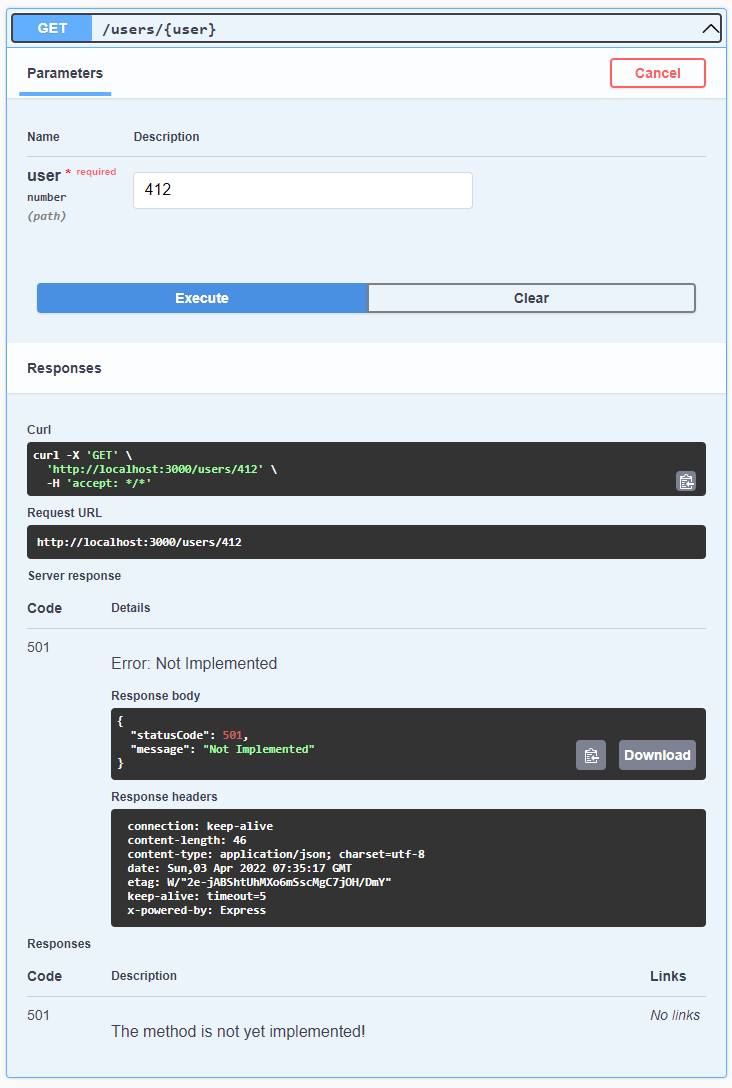
For swagger to work we also need to initialize it in main.ts:



If we try to run our project now and go to the endpoint /api, we can see the following:



Each of these points can be documented using decorators.

We can also test these endpoints: